

WHAT IS CLAIMED IS:

- 1 1. A method comprising:
 2 receiving configuration data that indicates which one of a plurality of memory
 3 controllers will support each of a plurality of clients,
 4 receiving data access requests from each of the plurality of clients; and
 5 routing each of the data access requests to one of the plurality of memory controllers
 6 based on the configuration data.

- 1 2. The method as in Claim 1, wherein the plurality of clients and the plurality of
 2 memory controllers are integrated on a single device.

- 1 3. The method as in Claim 2, wherein the single device is a semiconductor device.

- 1 4. The method as in Claim 3, wherein the semiconductor device includes a graphics
 2 controller.

- 1 5. The method as in Claim 1, wherein data access requests routed to the plurality of
 2 memory controllers are executed by the plurality of memory controllers in the order
 3 in which the data access requests are received.

- 1 6. The method as in Claim 1, wherein the plurality of clients include at least two clients
 2 having a common client type.

- 1 7. The method as in Claim 6, wherein the common client type includes one of the group
 2 of a two-dimensional graphics driver, a three dimensional graphics driver, and an
 3 audio driver.

1 8. The method as in Claim 6, wherein the step of routing includes routing access
 2 requests from a first client having the common client type to a first memory controller
 3 and routing access requests from a second client having the common client type to a
 4 second memory controller.

1 9. The method as in Claim 8, wherein the first memory controller executes a first portion
 2 of an access request and the second memory controller executes a second portion of
 3 the access request.

1 10. The method as in Claim 6, wherein the step of routing includes routing access
 2 requests from a first client having the common client type to a first memory controller
 3 and routing access requests from a second client having the common client type to the
 4 first memory controller.

1 11. The method as in Claim 6, wherein the step of routing includes routing a first access
 2 request from a first client having the common client type to a first memory controller
 3 and routing a second access request from a second client having the common client
 4 type to the first memory controller, wherein the first and second access requests are
 5 simultaneously pending at the first memory controller.

1 12. The method as in Claim 11 further comprising a step of prioritizing one of the first
 2 access request and the second access request based upon a predefined arbitration
 3 scheme.

1 13. The method as in Claim 12, wherein the predefined arbitration scheme is a round
 2 robin arbitration scheme.

- 1 14. The method as in Claim 12, wherein the predefined arbitration scheme prioritizes
2 access requests from the first client over access requests from the second client.
- 1 15. The method as in Claim 11 further comprising a step of prioritizing one of the first
2 access request and the second access request based upon a first predefined arbitration
3 scheme when the first and the second access requests are from clients having a first
4 common client type, and prioritizing one of the first access request and the second
5 access request based upon a second predefined arbitration scheme when the first and
6 the second access requests are from clients having a first common client type.
- 1 16. The method as in Claim 15, wherein a priority assigned to the access requests is
2 dynamic.
- 1 17. The method as in Claim 16, wherein the priority assigned depends on an identifier
2 within the access request.
- 1 18. The method as in Claim 16, wherein the priority assigned depends on an internal
2 timer.
- 1 19. The method as in Claim 1, wherein client requests are routed based on one or more
2 of: an address, a client identifier, client tag information, and data size.
- 1 20. The method as in Claim 1, wherein the number of requests routed to a memory
2 controller, from a particular client, is dependent on the data rate of the particular
3 client.
- 1 21. The method as in Claim 1, wherein the plurality of memory controllers are scalable.

1 22. The method as in Claim 1, wherein:
2 the step of receiving data access requests includes receiving a first HDTV stream and
3 a second HDTV stream; and
4 the step of routing includes:
5 routing the first HDTV stream to the first memory controller; and
6 routing the second HDTV stream to the second memory controller.

1 23. The method as in Claim 1, wherein one of the plurality of memory controllers is
2 dedicated for use in handling requests received from a high-data rate client, where the
3 high-data rate client requests a greater amount of data than other clients.

24. An apparatus comprising:

- a storage module having a memory location and an output port, the memory location to store data;
- a plurality of clients, each of the plurality of clients having a data access port;
- a router having a plurality of first-input ports coupled to the data access port of each of the plurality of clients, a second input port coupled to the output port of the storage module, a first plurality of output ports, and a second plurality of output ports, wherein the router is to route data at one each one of its plurality of first- input ports to its respective output port of the first or second output port based upon the data stored in the storage module;
- a first memory controller having a plurality of input ports coupled to the first plurality of output ports of the router;
- a second memory controller having a plurality of input ports coupled to the second plurality of output ports of the router; and
- a first arbiter having a plurality of input ports coupled to the first plurality of output ports of the router, and an output port, wherein the first arbiter selects one a data access request on one of the first plurality of input ports to be provide to the output port.

25. The apparatus of claim 24, further including:

- a second arbiter to receive a plurality of client requests from a plurality of clients, to route each of said requests to one of a plurality of memory controllers based on a programmable value; and
- a plurality of memory controllers to order client requests, and to deliver said ordered client requests to memory, at least two of the plurality of clients having a common type.

26. The apparatus as in Claim 24, wherein said first memory controller and said second memory controller include arbiters, said arbiters to order client requests.

1 27. The apparatus as in Claim 26, wherein said arbiters perform round robin arbitration
2 between clients having a common type.

1 28. The apparatus as in Claim 26, wherein said arbiters performing a first arbitration
2 between clients having a common type, and a second arbitration between clients
3 having different types.

1 29. The apparatus as in Claim 24, wherein said router routes a first client request from a
2 first client to a first memory controller, and routes a second client request from a
3 second client, the second client being the same client type as the first client, to a
4 second memory controller.

1 30. The apparatus as in Claim 24, wherein said router routes the client requests based on
2 one or more of: an address, a client identifier, client tag information, and data size.

1 31. The apparatus as in Claim 24, wherein the number of requests routed to a memory
2 controller, from a particular client, is dependent on the data rate of the particular
3 client.

1 32. The apparatus as in Claim 24, wherein said plurality of memory controllers are
2 scalable.

1 33. The apparatus as in Claim 24, wherein:
2 said router receives a first HDTV stream from a first client and a second HDTV
3 stream from a second client; and wherein
4 said router routes the first HDTV stream to a first memory controller and routes the
5 second HDTV stream to a second memory controller.

- 1 34. The apparatus as in Claim 24, wherein one of said plurality of memory controllers is
2 dedicated for use in handling requests received from a high-data rate client.

- 1 35. A method comprising:
2 receiving a first client request from a first video decoder;
3 routing the first client request to a first memory controller;
4 receiving a second client request from a second video decoder; and
5 routing the second client request to a second memory controller.
- 1 36. The method as in Claim 35, further including providing the first client request to a
2 first memory and the second client request to a second memory.
- 1 37. The method as in Claim 35, wherein routing is based on one or more of: an address, a
2 client identifier, client tag information, and data size.
- 1 38. The method as in Claim 35, wherein the number of requests routed to a memory
2 controller, from a particular client, is dependent on the data rate of the particular
3 client.
- 1 39. The method as in Claim 5, wherein the memory controllers are scalable.
- 1 40. The method as in Claim 35, wherein:
2 the first video decoder is an MPEG decoder;
3 the first client request is a first HDTV stream;
4 the second video decoder is an MPEG decoder; and
5 the second client request is a second HDTV stream.

- 1 41. The method as in Claim 35, wherein one of the memory controllers is dedicated for
2 use in handling requests received from a high-data rate client.